Implementation of the OpenURL and the SFX Architecture in the Production Environment of a Digital Library

Miriam Blake
Library Without Walls Team member
Los Alamos National Laboratory Research Library
meblake@lanl.gov

Abstract:
The Los Alamos National Laboratory Research Library was an early adopter of the OpenURL framework, implementing the SFX architecture live across multiple in-house databases and an extensive electronic journal collection housed both internally and at outside publisher/aggregator sites. Basic issues of reference linking such as appropriate copy, context-sensitive linking, and the need for standards in open solutions are mentioned. Use of OpenURL in a general framework and its incorporation into SFX and deployment in the larger scholarly information environment is discussed. The paper focuses on practical considerations in the implementation of SFX and OpenURL in an evolving production environment.

Background

Reference linking is generally described as the links between one information object and another [Caplan and Arms, 1999]. These links can be between a wide variety of information sources, such as references in a bibliography, citation references in a database, or links from less formal sources such as those found on informal websites. One of the most common, and most quickly addressed, forms of linking is between journal articles; the links between references at the end of an article to the referred article itself are maybe the most direct representation of reference linking. Because this is such an obvious manifestation of linking in a web-based journal environment there is increasing demand and co-operation from publishers to help create the mechanisms to support a large-scale linking infrastructure; the Digital Object Identifier (DOI) system is such a publisher based initiative. [Atkins, 2000]. Increasing use of the DOI and incorporation of it into linking systems is happening rapidly. Additionally, there are new mechanisms being developed which are creating and beginning to standardize a large-scale linking infrastructure beyond just journal article citation linking, such as OpenURL and link-server architectures like SFX. These evolving elements begin to address several different pieces of the larger picture, which create a system that can be characterized as dynamic or open and context-sensitive. These characteristics are the groundwork for linking systems and will be covered here.

Problems with existing methods:

Early systems that created links between scholarly information usually employed architectures that were static in nature; links were precomputed and built into a linking database, so when users clicked on a link it was more or less a direct identifier to the location of the linked article. This was true for both early publisher initiatives and digital library collections like the one at the Los Alamos National Laboratory (LANL) Research Library. However, this framework was shown to be inadequate for large-scale linking [Van de Sompel and Hochstenbach 1999a]. The main issues not handled by static links can be summed up:

- Limited in scope: these links, both in content delivery and action sphere, were limited basically to local or locally licensed content bases. They basically reside only in the domain of the authority providing the links.
- Maintenance intensive – often enabling new links into these databases requires time and resource intensive machine processing

One alternative to static systems is to create links “on-the-fly” from metadata within the source object. This relies on two general assumptions: the presence of metadata elements from which to calculate linking information, and the presence of a calculable “link-to” syntax for the target resource. Currently there is a lack of adopted standards in this area, although many publishers are now creating calculable internal syntaxes, which they make available to institutions desiring to create links to their material. Examples are the APS Link Manager (http://publish.aps.org/linkfaq.html) and the Blackwell Synergy Resolver (http://www.blackwell-synergy.com/journals/publisher custom/bsl/resolver.rtf).

Recent initiatives have also been directed at creating reliable but “authority independent” links between information objects. The DOI is an attempt by publishers to create a large-

scale system of named entities that can be used as persistent identifiers for any manifestation of a work. The idea of the DOI is to create a unique identifier that can be used for every mention of a work, no matter where that mention is located. If a DOI is known by a resource presenting a link, it can be redirected (via the CNRI Handle system – see http://www.handle.net/) directly to the work. In a related initiative, many STM publishers are depositing metadata into the CrossRef system, a specific application of the DOI for linking citations. CrossRef links DOIs to tagged metadata for uses such as providing interoperability between digital materials (such as including a film clip in an article). Another use for the CrossRef metadata of special interest here is to provide extended services to overlay systems such as link-servers. However, DOIs are not yet commonplace in most citations, databases, and articles on the web. Additionally there will continue to be the historical issue of discovering links to older materials if DOIs do not get assigned retrospectively to already published materials or added to older commercial citation databases.

The other major issue that has been identified as necessary for a useful linking infrastructure is user context sensitivity. Previous linking methodologies do not take into account the user's context for actions such as enabling redirection to appropriately licensed services such as full-text delivery or disabling links to items outside of the user's available content range. [Van de Sompel and Beit-Arie 2001]. Users who want to follow a link need to be identified in some manner for two main reasons:

1. Determination of the "appropriate copy." This is a critical problem, which is being rapidly addressed by both the OpenURL and DOI communities [Beit-Arie, et. al. 2001]. The basic problem is that default link locations generally supplied by publishers do not take into account the numerous locations from which licensed content might be correctly appropriated by a given user in the distributed web environment. In order to redirect to the appropriate copy of a work, some user information must be passed to the service providing the link, understood by that service as user location information, and redirected to another service, which could interpret where an appropriate copy is located. Again, this interpretation is a function of the link-server in the current model.

2. Presentation of extended services. As detailed in work by Herbert Van de Sompel and his colleagues at Ghent [Van de Sompel and Hochstenbach 1999b], navigation in a web-based scholarly information environment should and can extend well beyond the notion of a classical reference link. With the presence of even brief citation metadata, users can be offered links into a large array of other available web services beyond traditional full-text retrieval, such as citation database searches, general web searches, online bookstores, online catalogs, etc. However, this relies on some user identification in order to present access to appropriate and licensed services; otherwise such services could overwhelm users with irrelevant options and links to unlicensed/inaccessible content.

OpenURL and link-servers as part of the solution

As one part of the developing framework, OpenURL adds the first step of a standardized solution to the linking problem. Accepted as a fast-track work item by NISO for accreditation as an ANSI standard, OpenURL is a methodical approach to transporting information needed to deliver customized, relevant and appropriate services. An "actionable URL," OpenURL is a flexible component that is inserted into an information resource as a hook to transport information to the user's link-server. The hope is that this will become a fairly simple and standardized mechanism that takes the burden of linking off information providers and allows it to be handled by individual libraries where it can be customized and maintained more appropriately. In addition to relieving information providers of the infrastructure issues involved in maintaining links for all their customers, OpenURL provides the promise of integrating information resources on a large and comprehensive scale, which could add great value to any single service.

OpenURL handles the following:

1. Context-sensitivity. This means determining the existence of a link-server/institutional service component, and its subsequent location. One way this is commonly done by a cookie delivered via a script installed in the same domain as the information resource. It becomes the "BASE-URL" portion of the OpenURL. (See the draft OpenURL specification a http://www.sfxit.com/openurl/openurl.html for details). Alternatively, some Information Providers are choosing to extend the user profiles they already have in place to push information into the OpenURL BASE-URL.
2. Metadata transport. OpenURL then transports metadata or keys to access metadata for the object for which the OpenURL is provided to the link-server – the target for the OpenURL.

OpenURL contributes an open standard to the competitive world of information providers who clearly need to enable reference links and extended services in a decentralized way. This means, however, that the decisions about appropriate and relevant services for a given source are shifted to the institution providing the services directly to the users, i.e. the library. This is where the link-server comes into play. The institutional service component or link-server technology is the newest "killer app," being developed in tandem by organizations and companies to allow libraries and information providers a way to choose which resources to link to and how to present these links to users. This technology is new enough that general terminology for it is not standardized; it is referred to as an "institutional service component (ISC)," a link-server, or an OpenURL resolver, to name a few. It will be referred to as a link-server here. SFX, marketed by Ex Libris Information Services Division, is the first production version of such an application; it is a link-server which holds a "rules" database of library-specific content and link preferences. By using the OpenURL framework, the SFX software allows users to link between a variety of web resources, such as citation databases, journal articles, web search engines or document delivery services.

The following (Figure 1) illustrates the general flow of the OpenURL framework.

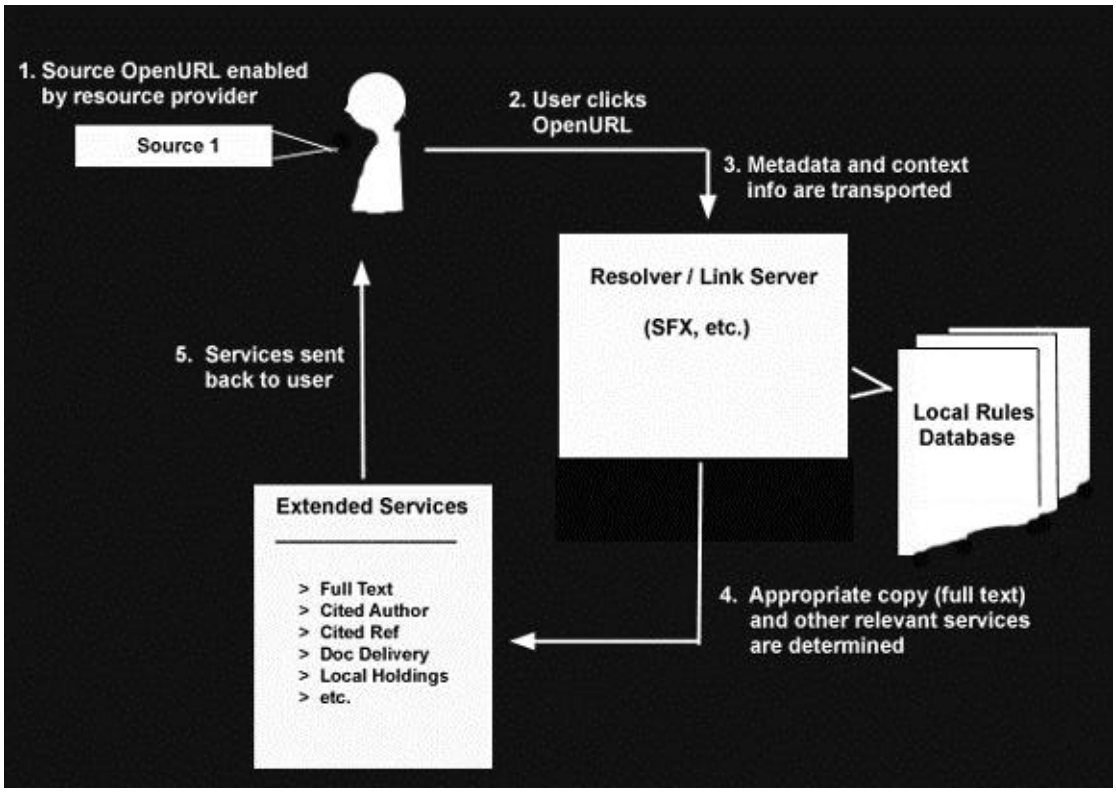

Figure 1 – flow of OpenURL through a link-server

Implementing SFX at LANL

The Los Alamos National Laboratory (LANL) Research Library was an early implementer of SFX. Originally involved in the prototype work on SFX with Herbert Van de Sompel and the University of Ghent [Van de Sompel and Hochstenbach, Oct. 1999], LANL recognized the ideas underlying SFX as a solution to the growing problem of linking, both between in-house collections and from these collections out into the wider scholarly collections on the web. The LANL digital library collections consist of eight locally loaded major commercial databases (including Biosis, INSPEC, Engineering Index, and a locally modified version of ISI's Web of Science called SciSearch at LANL), containing over 56 million citations, and a local repository of over 2.5 million journal and conference electronic articles from 5 major publishers. Additionally, LANL subscribes to electronic content from 175 other publishers and 90+ external databases. Prior to SFX, linking at LANL was limited to two basic forms: links from citations in the databases to locally held electronic articles, and links from the citations to external articles where a clear "link-to" syntax could easily be discovered and calculated. In both cases, a static database was kept in which rules for link calculation was manually determined, and specific journal information (such as ISSN, link calculation "type," and starting/ending subscription dates) were entered manually by staff. Whenever new journal ISSN's were added or deleted from the linking database, a rebuild of the affected collections was required to update the citation databases.

The prototype work on SFX required LANL to OpenURL-enable all the in-house databases. Because the databases were in local control, the operation was simple and did not require external vendor buy-in. (The University of Ghent, who partnered in the project, did use external vendors such as SilverPlatter and Ex Libris for the test). The decisions made during the prototype regarding how to OpenURL-enable the databases are still applicable in the current production version of SFX at LANL. The following process was used to OpenURL-enable the LANL databases:

1. The record key is the only metadata element passed to the OpenURL behind the "SFX Button." Once a user clicks the SFX Button, the record key is used to retrieve the metadata from the source. The full metadata record is then passed to the SFX server for presentation of services.
2. The local tagged output program was modified to add additional metadata that might be useful to create SFX services. The tagged format of the complete citation comprised the metadata returned to SFX when the OpenURL is clicked.
3. Javascript for the "CookiePusher" (http://www.sfxit.com/openurl/cookiepusher.html) was added to initial General Search entry screens for each database. This forces the SFX Button to appear for all users in the LANL domains.

Note: the LANL library serves not only LANL users, but also has external customer sites who contract for services, but who do not have access to SFX.

The initial testing of the SFX prototype in 1999 proved successful. In early 2000, Ex Libris purchased the technology from Ghent and subsequently invited LANL to participate in the Beta testing of the software. In November 2000, LANL went live with an early production version of the SFX software, marketed at LANL as "LinkSeeker", after initial feedback that SFX was a confusing name for users. Large-scale implementation of the production software has obviously provided a much broader experience for LANL than simply testing the prototype. The following are observations from the LANL experience implementing and maintaining SFX.

Terminology

One of the biggest difficulties in initial implementation was learning the new terminology that surrounds SFX and the linking technologies it embodies. The following terms are used in conjunction with the description of SFX:

Source: the information source where the user begins, such as a citation database or journal article with references. The metadata which SFX acts upon generally comes from the source. An SFX source must have implemented OpenURL. Note that a source, such as a database, can also be a target.

Services/Extended services: Options, decided by each library/SFX installation, which are presented to the user after they click an SFX Button. These can include an option to retrieve a full-text article, search any one of the author names in another database, do a web search for words from the source citation, do a document delivery request, etc.

Target: Where the user ends up after clicking on one of the services – the full-text article, a web page, a document request form, etc. Targets in the SFX database are databases, journal publishers or any other "parent" level record that includes information universal to all objects associated with that target, such as the publisher website URL.

Objects: Items in the SFX database, such as a specific record for a journal.

Object portfolio: The collection of linkages and services for each object and target, such as "getFullText" or "getAuthor." In combination with the threshold, the portfolio determines which services will be offered for each source record.

Thresholds: Limitations for a specific object. For example, the starting date, volume, and issue for accessing the electronic version of a journal, as well as required information needed to construct the "link-to" syntax URL, are all part of the threshold information.

Software Installation

LANL chose to install the SFX software in-house instead of going with Ex Libris as an ASP. The software runs on an Intel Pentium III, 750 GHZ with dual CPU, 1 GB memory and 1GB disk. It is a standalone Linux system running only SFX.

The SFX software installation is highly scripted, and the process installs an Apache server, MySQL, and Perl and Java libraries in addition to all the SFX code and directory structure. The installation process is well documented and ran smoothly. It is, however, a little disconcerting for users with a highly technical background to have all the Unix level installation/configurations done in such a scripted fashion. Some local configuration is necessary after the scripts have completed, but again, it is well documented.

SFX KnowledgeBase – setup

The heart of the SFX software is a core MySQL database called the "KnowledgeBase." This database generically holds all source and target objects which come with the installation. For example, it holds all rules information about the target Academic Press, and all journal objects associated with Academic Press – all the ISSN's, threshold information for each journal, etc. The installation comes with a global set of data – theoretically, all journals and all publishers that have electronic content and could be set up as targets, and all vendors who have OpenURL enabled their sites to be sources. Additionally, the software comes with specific parsers as needed for calculating peculiarities of a given publisher link-to syntax or for parsing the metadata coming in from an OpenURL enabled source as needed to create services.

The SFX model works on the premise that a "local instance" will be created by choosing and/or customizing sources and targets from the default set provided in the "global instance." After installation, the first step is to decide which sources, targets, and services will be offered by the local institution via the local instance. At LANL, a team was formed consisting of 6 people from different departments across the library. Areas of knowledge included Reference, Customer Service, electronic journals, and database metadata. It should be noted that 5 of the 6 team members were librarians with no IT skills; only one IT staff member was included on the team. The majority of work in setting up the local instance is done via a web interface to the MySQL database, and does

not require technical expertise. The technical work is needed mainly if the parsers, written in Perl, need to be changed or new parsers need to be created.

Initial meetings introduced the team to the terminology and basic structure and use of the SFX software. It was quickly decided that we would initially concentrate on the local databases as our sources (those that were already enabled as part of the prototype work). Because all initial sources are under local control, the source parsers had to be created locally. This was usually a matter of taking a tagged element and reading it into the corresponding SFX "generic request object" element. Metadata idiosyncrasies of each database were taken into account. (Later when outside sources were added, discrepancies/variations in metadata structure from commercial vendors were also analyzed and parsers were altered as necessary).

To decide which targets/services to implement, there were a couple of considerations:

1. Which services would LANL customers use the most? Historical feedback and discussion of user needs were relevant. Additionally, before going live with the production version, a focus group would be set up to provide feedback.
2. Which targets were already included in the SFX product? It was determined that only services which already had target parsers included in the product would be considered for initial implementation, unless otherwise determined critical to users.

Full-text was the most obvious service to include initially. Since the local source databases consisted primarily of citations to journal articles, this immediately created an SFX service for a large number of records in each database. Although LANL already had the static links built into the source databases, it was clear that SFX could offer more full-text links because of its dynamic nature. If a new subscription was added, or a wider range of electronic full-text became available from a publisher, the corresponding full-text links would immediately show up in the SFX services screen once they were added to the KnowledgeBase – no rebuilding of the databases required. However, full-text also required the most concentrated effort by the team. Because a large number of journals are locally loaded at LANL, local holdings had to be checked against the defaults in the global instance, and modified as needed in the local instance. Additionally, the default location information for these locally loaded journals needed to be change to point to the LANL repository instead of the default publisher location. For electronic journals outside of the LANL repository, the team chose to divide the workload by publisher and test each default publisher setup by making sure the links resolved to the publisher websites (and directly to the article level where possible) as expected. This uncovered some errors in the target parsers supplied with SFX, which were subsequently revised by Ex Libris. The workload involved was fairly heavy for team members during this phase of the setup, but this was considered useful as both because a comprehensive inventory of electronic holdings to which we are linking was also completed. This database can be queried independently of the SFX software for other local projects.

One unique problem presented by adding the full-text service to SFX was the duplication with the static database links already in place. User feedback has indicated that "one-click" access to the full-text was preferred by users, but SFX requires two clicks (first the SFX Button to get the services screen, and then another click to go to the full-text target). The decision was made to keep both the full-text link icon and the SFX icon for each citation. (Figure 2).

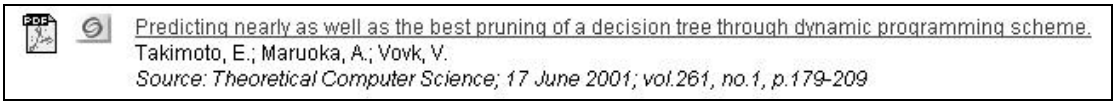

Figure 2 – database citation record with static link and SFX button.

The longer term plan is to replace the static database linking system with a dynamic call into the SFX link-server for the full-text link via an OpenURL syntax, allowing the user to go directly to the full-text and thus relieving the need for database rebuilds as required by the static system.

In addition to full-text, the other services added for the initial production version of SFX are as follows:

- *Document delivery*: In order to integrate the SFX document delivery service with the local cgi form already in use at LANL, the document delivery parser had to be modified to pass the appropriate metadata from the citation record into the existing form. Offered only when full-text is not available from existing LANL sources, this service automatically fills in a document delivery request with the relevant citation metadata. It has proven to be a high use SFX service.
- *Cited author/cited paper searches*: It made sense to offer the popular cited searching capabilities of SciSearch at LANL (the ISI dataset) as an SFX service. From each source database, any author listed in the source citation is available from the SFX service screen via a drop-down box. Journal information is parsed and combined with author information to create a cited paper search. Since this is basically a search from a local source into a local target, both local source and target parsers were written to accommodate it.
- *Author search across all databases*: LANL uses a locally developed multi-database search engine called "FlashPoint" to search all local databases simultaneously. Offering a service which would send an author search to FlashPoint (again, any author listed in the source citation would be available for searching) was a way maximize SFX by using the local databases as *targets* this time instead of sources.
- *Local OPAC search*: One of the common needs link-servers can easily address is checking local catalogs for print holdings. To implement this for a standard Z39.50 based OPAC, a Z39.50 compliant server must translate the query into the Z39.50 protocol. The Zeta Perl package is part of the generic SFX installation. The query syntax into the local OPAC (Geac Advance) was defined, and source metadata records containing either ISSN, ISBN, or Technical Report Number fields are translated via the Zeta Perl module and sent into the OPAC to check for local holdings. (Citations without one of these fields do not present a local holdings service in SFX). If holdings are not found, a link to the document delivery form is presented. It should be noted that the local Z39.50 compliant webserver provided by Geac, GeoWeb, is *not* used to present the results because external query does not allow for a session id to be established in the native GeoWeb interface. Z39.50 results are returned and presented from the Zeta Perl interface modified to LANL specifications.
- *Author E-mail search*: A target parser which sends an author name to a web email search service is included in the standard SFX package, so it was included as an added service.

- *PubMed genome search*: The Biosis database includes genomes as a field in the metadata. A parser is included in the standard SFX package which will send a genome search to the NIH PubMed Genome database. This is an excellent creative use of SFX and shows the flexible use of different types of metadata to generate additional services.

As each of these services was added, the team tested with each database to ensure expected results.

Finally, before going live with SFX, a user focus group session was held to make sure the product was understandable and would be useful for "real users." The feedback resulted in some local customization of the services screen. The major change was to use radio buttons instead of drop down boxes when multiple databases could be used for a given search. (Figure 3).

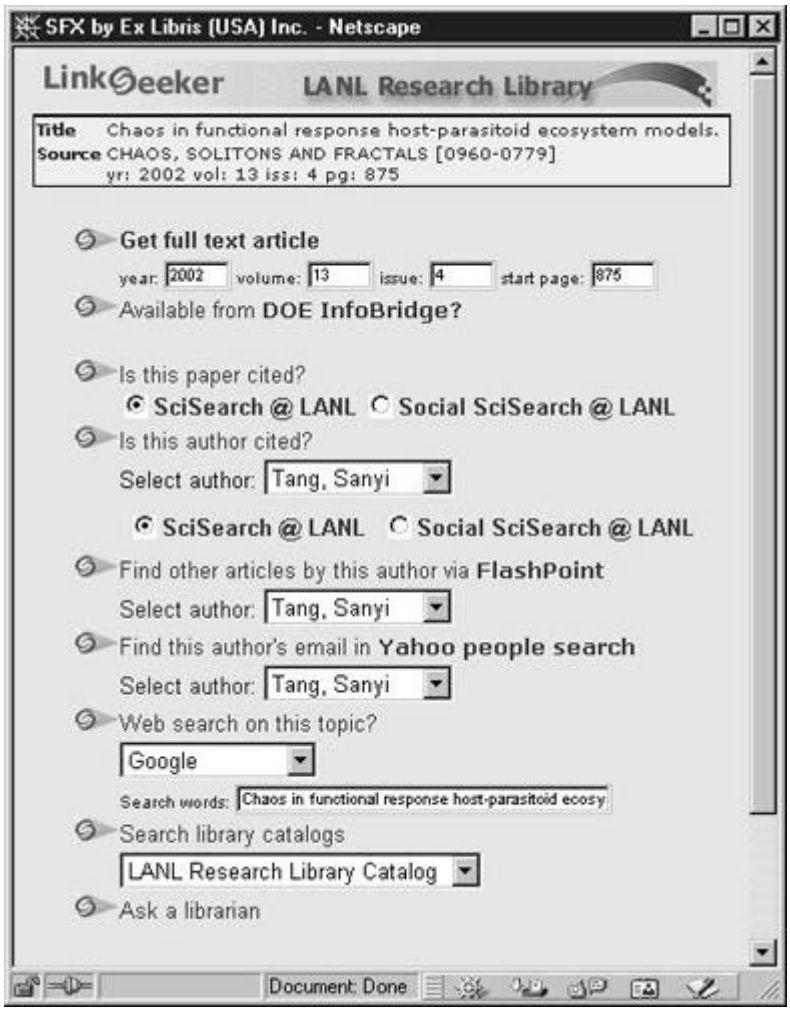

Figure 3 – LANL's SFX services screen

Users suggested some new services but felt the services being offered were also useful. Feedback also seemed to indicate that SFX is a product that is best understood by *being used*. It is a technology that integrates information resources and is difficult to visualize outside of those resources. This gave our marketing team something to think about.

SFX went into production in November 2000, after about 6 months of testing (initially using Beta software). After going into production, LANL has maintained a "test instance" which mirrors the live local instance, but is accessed by setting a different cookie which points to a test KnowledgeBase and a different set of configuration directories in the SFX system. This allows for ongoing testing as new sources, services, and targets are added.

Issues and ongoing work

The number and detail of services that can be presented in SFX is directly related to the metadata that is pulled from the source record. For example, a link to journal full-text requires an ISSN to be present in the source metadata. Because the implementation of SFX at LANL initially used only in-house (and thus locally controlled) sources, LANL did not have to immediately grapple with issues of inconsistent metadata coming in from outside sources. However, it was challenging in its own right to work on metadata issues which arose from the internal databases. One of the most complex issues is the handling of author names. Each database has its own variations in author name formatting, such as flipped lastname / firstname, full first names vs. initials only, etc. So a fair amount of time was spent massaging author names to provide as much consistency as possible between different sources. In summer 2001, LANL participated in the prototype experiment with CrossRef, the International DOI Foundation, CNRI, Ex Libris, and several other institutions to address the appropriate copy problem [Beit-Arie, et. al., 2001]. As part of this experiment, CrossRef passed back metadata for reference citations linked to DOIs. For a user who had access to a link-server, such as SFX at LANL, the metadata was used to present extended services. Since CrossRef collects only the basic metadata necessary for matching references to article DOIs, the richness of the returned metadata was often limited, especially when compared with that returned from locally controlled databases.

Although the SFX KnowledgeBase does become a fairly comprehensive source for a library's electronic collection inventory, it is not static or complete once all accessible objects are activated. Specifically the following issues are of ongoing importance:

1. Library journal holdings change continuously. Subscriptions are added or removed, and these must be maintained by the library staff in the local instance of the SFX KnowledgeBase.
2. Publisher websites change without notice. URLs, access methods/restrictions etc. can change and there is often notification until a user reports a broken link.
3. Ex Libris does not, understandably, have subscription access to all journals at every website. This means that the global setup may come with only TOC access, or without correct settings for accessing the article level, so parsers should be tested as they are activated.

Librarians have known for some time that maintaining an electronic collection is a lot of ongoing work. Ex Libris has committed to maintaining the KnowledgeBase with current information. A reporting utility is included for libraries to help in this monumental task by reporting problems as they find them. Updates to the KnowledgeBase are planned as a regular part of SFX. Additionally, Ex Libris will add new parsers as more commercial sources become OpenURL-aware and as more targets are available for linking.

More services have been added since SFX went into production, including a general web search which sends words from the title out to a user-selected search web search engine and an “Ask a Librarian” service which allows for free-text email to be sent to library reference staff. Additional journal objects and new publisher targets are added regularly.

Conclusion

The world of reference linking and extended services adding value to web-based scholarly information is growing at a rapid pace. Work progresses on standardizing the OpenURL (http://www.niso.org/commitax.html). Of interest is the view being taken by the NISO OpenURL committee that the standard should be approached as a more generic mechanism for making identifiers and metadata available to service components beyond the scholarly web and into the larger web environment, and the subsequent beginnings of this research [Van de Sompel and Beit-Arie, 2001]. The DOI is being experimented with or assigned to works by over 100 different publishers, and is becoming a basic “e-commerce building block.” Link-servers are now being developed by other major information systems vendors, giving Ex Libris and SFX some competition. Endeavor Information Systems has announced plans to have a link-server available by the end of 2001. Openly Informatics 1Cate uses OpenURL to link to ejournal holdings and offers other add-on features of link-server functionality piece by piece. UKOLN's OpenResolver service demonstrates yet another link-server currently set up to test the OpenURL functionality. These and other similar activities seem to point to link-server technology as a critical component of the emerging reference linking infrastructure, and an area where competition and creativity will define future developments.

References

Atkins, Helen, et al., ‘Reference Linking with DOIs’, *D-Lib Magazine*, 6. 2 (February, 2000), http://www.dlib.org/dlib/february00/02risher.html

Beit-Arie, Oren, et. al., ‘Linking to the Appropriate Copy : a Report of a DOI-Based Prototype’, *D-Lib Magazine*, 7. 9 (September, 2001), http://www.dlib.org/dlib/september01/caplan/09caplan.html

Caplan, Priscilla and Arms, William Y., ‘Reference Linking for Journal Articles’, *D-Lib Magazine*, 5. 7/8. (July/August, 1999), http://www.dlib.org/dlib/july99/caplan/07caplan.html

Van de Sompel, Herbert and Beit-Arie, Oren., ‘Open Linking in the Scholarly Information Environment Using the OpenURL Framework’, *D-Lib Magazine*, 7. 3. (March 2001), http://www.dlib.org/dlib/march01/vandesompel/03vandesompel.html

Van de Sompel, Herbert and Hochstenbach, Patrick., ‘Reference Linking in a Hybrid Libary Environment, Part 1: Frameworks for Linking ‘, *D-Lib Magazine*, 5. 4 (April, 1999), http://www.dlib.org/dlib/april99/van_de_sompel/04van_de_sompel-pt1.html

Van de Sompel, Herbert and Hochstenbach, Patrick., ‘Reference Linking in a Hybrid Libary Environment, Part 2: SFX, a Generic Linking Solution’, *D-Lib Magazine*, 5. 4 (April, 1999), http://www.dlib.org/dlib/april99/van_de_sompel/04van_de_sompel-pt2.html

Van de Sompel, Herbert and Hochstenbach, Patrick., ‘Reference Linking in a Hybrid Library Environment, Part 3: Generalizing the SFX solution in the "SFX@Ghent & SFX@LANL" experiment’, *D-Lib Magazine*, 5. 10 (October, 1999), http://www.dlib.org/dlib/october99/van_de_sompel/10van_de_sompel.html